ARMY RESEARCH LABORATORY

CyFall:
A Cyber-Network Game Scenario

**by Renée E. Etoty, Robert F. Erbacher, Steve Hutchinson,
Richard Astrom, Garrett Payer, and Raymond Chang**

ARL-TR-7024 **August 2014**

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory
Adelphi, MD 20783-1197

ARL-TR-7024 **August 2014**

CyFall:
A Cyber-Network Game Scenario

Renée E. Etoty, Robert F. Erbacher, Steve Hutchinson,
Richard Astrom, Garrett Payer, and Raymond Chang
Computational and Information Sciences Directorate, ARL

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED (From - To)
August 2014	Final	09/2012 to 09/2013

4. TITLE AND SUBTITLE
CyFall: A Cyber-Network Game Scenario

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Renée E. Etoty, Robert F. Erbacher, Steve Hutchinson, Richard Astrom, Garrett Payer, and Raymond Chang

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
ATTN: RDRL-CIN-D
2800 Powder Mill Road
Adelphi, MD 20783-1197

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TR-7024

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

User studies are a critical component in the evaluation of the user interface components of tools where direct algorithmic performance comparison is not applicable. Optimally, these studies would use domain experts as the study subjects to maximize the relevance of the results. However, often insufficient numbers of domain experts are available for the user studies in which case nonexperts must be used as substitutes. The challenge is to be able to sufficiently engage the nonexperts, such that they can adequately perform in the study to acquire meaningful results. This report examines the development of a conceptual game (gamification) that would aid nonexperts in identifying with their role and the tasks they must perform such that multiple presentations of the underlying data can be effectively evaluated.

15. SUBJECT TERMS
design application, user study, gamification

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Renée E. Etoty
Unclassified	Unclassified	Unclassified	UU	36	19b. TELEPHONE NUMBER (*Include area code*) 301-394-1835

Contents

List of Figures

Preface

This report sets forth the design goals and requirements for the software, documents the design and development for the cyber game, and identifies the key details that are required to build the final scenario. CyFall, when completed, will be used for multimodal visualization tasks to engage players in pattern-matching, cognitive, and predictive activity involving a cyber-security scenario. This particular scenario will display output typically generated by a generic network intrusion detection system (NIDS) using at least three distinct presentation methodologies. An established paradigm associates the nodes in a graph to computers (i.e., network interfaces or Internet Protocol [IP] addresses in a network). The edges represent either actual communications, or latent/probable communications between associated nodes. This study focuses on brain activity correlated to this pattern-matching activity during a simulated analysis session similar to that of network analysts in cyber security. We ultimately aim to identify key components to visualization of cyber-security scenarios that make it more apparent to players that related attributes are in fact related; this is anticipated to enhance player performance with respect to both speed and accuracy.

INTENTIONALLY LEFT BLANK.

1. Background

There is a need to protect one's organization from attacks and attempts to intrude in the organization's system mounted by hackers and crackers, often using network connectivity as the initial attack vector. A hacker is someone who tries to break into computer systems (*1*). This type of person is likely to be a proficient programmer or engineer with sufficient technical knowledge and understanding of weak points in a security system. A cracker, on the other hand, is someone who breaks into someone else's computer system. The cracker has bypassed passwords, licenses, and intentionally breached the computer's security (*2*).

1.1 Intrusion Detection Systems (IDS)

The use of intrusion detection (ID) is a solution to the prevention of unwanted attacks and intrusions on computer network systems. All methods of ID involve the gathering and analysis of information from various areas within a computer or network to identify possible threats posed by hackers and crackers inside or outside of the organization. There are two main approaches to defending and protecting the organization: host-based and network-based ID systems (*3*).

Host-based intrusion detection systems (HIDS) rely upon features and observations that are local to monitor hosts such as file system activity, central processing unit (CPU) utilization, and disk space utilization to detect malicious activity. Such applications include a firewall, antivirus software, and spyware detection programs (*3*). A firewall is a software or hardware-based network security system that is capable of inspecting or filtering incoming and outgoing network traffic. The firewall (*4*) establishes a barrier between trusted and untrusted communications by examining the network packets against rules before deciding to forward them to its destination. Antivirus software is a category of defensive solutions that attempt to identify malware predominantly using signature-based approaches (*5*). Malware is malicious software that includes viruses, Trojans, key loggers, hijackers, dialers, and other variants that can vandalize or steal content from the computer. Spyware programs (*6*) are any technology that aids in gathering information about a person or organization without their permission and knowledge. In a typical use that addresses these threats, HIDS is initially and usually deployed in monitor-only mode (*7*). A HIDS approach monitors the system integrity, application activity, file changes, host network traffic, and system log files.

Network intrusion detection systems use antithreat software that relies on observations gathered mostly by sensors at specific points on the network to capture traffic between the outside environment and inside the organization's segment of network that needs protection (*3*). NIDS tools are security systems that monitor computer systems and network traffic and analyze that traffic for possible hostile attacks originating from outside the organization and for system misuse or attacks originating from inside the organization (*8*). Detected attacks can include a

broad spectrum of malicious activity from attempts to penetrate a computer system to attempts to gain unauthorized access to a network (*9*). The NIDS displays evidence in the form of "alerts," which are messages to the player indicating that some detection function has been triggered, or some threshold of activity has been observed and is displayed for consideration. The tools (a.k.a. "sensors") look at the network traffic and issue some messages that suggest an attack may be occurring. These tools are not perfectly accurate, and while an alert will often accurately indicate the presence of an attack (a True Positive [TP]); they may also emit erroneous alerts when there is no malicious activity present (a False Positive [FP]). Each alert is evidence of a possible intrusion incident and the analyst must use pattern matching, intuition, and reasoning to distinguish a TP from an FP. This reasoning often involves the consideration of two or more alerts in combination to detect a significant threat when they share some common attribute value. Some intrusions will only present one alert. Others may present up to five or more correlated alerts with corresponding threshold activity alerts.

1.2 Visualization in Cyber Security

Visualization has a history of being nondeployable, ineffective, and obfuscating—especially for the analyst, our end user. The overall goal of using visualization tools and techniques is to integrate them with interaction techniques effective for large-scale databases to analyze the data and identify sophisticated attacks within the arriving data (*10*). Therefore, the design of visualization techniques for the exploration, analysis, and situational awareness of network events has become a significant focus of researchers as they attempt to deal with the sheer volume and complexity of the data (*11*). This has resulted in two cognitive task analysis (*12, 13*) examining the needs and requirements of network analysts and managers. In (*12*), the study used event-related functional magnetic resonance imaging (FMRI) to study the pattern of activation during four distinct stages in the performance of the Wisconsin Card Sorting Task (WCST). Ellis and Dix (*13*) conducted an explorative analysis on user evaluation studies that use information visualization. They found that an empirical evaluation of visualizations alone is methodologically unsound because of its generative nature. Their results do show that empirical evaluations used in conjunction with reasoned justification leads to a more reliable validation of the visualization. This direction of research has resulted in the development of enumerable visualization techniques. The entire community, VizSec (*14*) has been formed around the research task of visually analyzing and monitoring network data that is usually reviewed at their yearly conference.

Visualization for intrusion detection can help a security administrator to recognize abnormal behavior in an intuitional manner. Visualization of intrusion detection can enable better analysis and response because an intrusion is recognized intuitionally. Therefore, it can overcome alert flooding. Most ID methods with visualization are anomaly-based detection methods and visualize audit data rather than the alert itself (*15*). The host-based visualization method for intrusion detection is to learn normal states of commands or programs that is achieved by the user and compares audit data with profiles for visualization.

Network-based visualization method used for intrusion detection expresses the source address, destination address, port number, and so forth of the network's packets by visual graph (*15*, *16*). They detect an intrusion when an attack differs from graph characteristic with normal state, and extract diagnostic features of attack for embodying anomaly detection. However, these methods do not visualize alerts, but visualize audit data. This is useful for detecting attacks that emit much traffic such as a distributed denial-of-service (DDoS) attack (*17*). This method does not offer clear features for attacks that emit little traffic (*18*).

1.3 Data Fabrication

Due to the sensitivity of capturing data from the U.S. Army Research Laboratory (ARL) network and computing infrastructure for experimental purposes, it was necessary to fabricate and simulate data for the cyber-game scenario. In order to compose a representative dataset for the game, we needed to define parameters that shape the general look and feel of "world or universe" for the player to operate. Therefore, in a brainstorming session, we raised some initial questions:

1. What is the nature of visualization study in which the game is needed?
2. What are the requirements and objectives for the dataset to support the game?
3. What are the features of observable network traffic and how can they be visualized?
4. What parameters of the network dataset can convey value graphically?
5. What is the scenario for the game and will it be created?

Formulated blueprint:

1. The nature of the visualization study is to look at brain activity associated with pattern-matching activities exercised via a tabula, graphical, and hybrid displays.
2. The needed requirements and objectives for the dataset to support the game are:
   a. Create fairly authentic patterns and concepts;
   b. Create authentic display(s) and player environment(s);
   c. Fabricate realistic data to support patterns that can be recognized and labeled within approximately 30 s of manipulation;
   d. Dataset should be derived from real-world observational instance records unified by a plausible scenario.
3. Features of an observable network traffic are:
   a. Nodes.
   b. Links.
4. Features that can convey value graphically are:
   a. Position.

b. Proximity.
c. Color.
d. Shape.
e. Area and pattern of glyph.
f. Directionality.
g. Textual labeling/tooltip.

5. The scenario for the game will be created with the following:
   a. Develop a storyline that cover some background in the domain.
   b. Generate friendly and unfriendly agents and name the conventions for the entities.
   c. Generate pseudo attacks, the resulting “alert records” that should be observed.
   d. Identify the role of agents.
   e. Specify injected “noise” and benign alerts.
   f. Make threats proportional to noise.
   g. Conduct algorithmic mapping of third-party traffic into the scenario for background and normal traffic.

In this application, we determined that node identity, which correlates to a node’s Internet Protocol (IP) address, would provide all of the required linking information needed to define the topology of the scenario graph network. This preserves authenticity with respect to the cyber-security domain prescribed by the scenario and facilitates incorporation of open-source Snort alert rules (*19*) into the game scenario. Snort alerts do form smaller graphs that are linked or unified by their message category assertion and diversified by the strings, as well as IP addresses to which each is sensitive. The scenario designer determined an attack scenario of many external nodes attacking a smaller number of friendly peer nodes. An IP address for each node was selected from a synthetic grouping of addresses where the higher-order address octet values were common to a presumed “country” of origin. Node linkage was then accomplished by essentially test-playing each attack and inserting the proper IP address of the attacker and target nodes into mock “alert records” to constitute the scenario. As a result, we have a list of alerts representing a number of distinct steps for a number of attacks. We used 10 attacks for this particular case. The scenario designer then established observation timestamps for each such alert to preserve the expected order-of-arrival for each attack type. Next, they interleaved traffic of all attacks into the scenario interval, which was established at 1 h. The designer then added alerts for benign traffic that function as typical false alerts or FPs, to provide an adjustable set of “noise” in the alert dataset. Initially, this process created over 300 such false alerts. Later in the visualization process, we determined that for the nature of the visualization study, this was far too many false alerts to deal with, so these were reduced to 10 false alerts per stage, per player. In addition, we provided a new spreadsheet that contains added columns to implement new entity naming, country attribution of source and destination, and removes all but 10 of the false alerts as noted above. Changes were saved in a comma-separated values (CSV) file and used to generate the following updated visualizations for the game.

2. Game Overview

2.1 Game Concept

We have designed a cyber-network game scenario to illustrate a typical analysis session comparable to the analysis that cyber-security analysts perform. The goal of the game is to present similar types of cyber-security activity to test the player on their ability to discriminate the patterns, using both a representative textual display as well as two more graphically oriented displays. The game environment presents the player with displays that emanate from a network intrusion detection system (NIDS). Three visual representations appear before the player during the game: a tabular display, a parallel display, and a node-link display. The game runs the same sequence of attack scenarios in a randomized fashion for each of the three displays presented to the player. Each of the three visual displays will have three modules: Introduction, Exercise, and Results. The player acts in the role of an analyst, examines the evidence available from the NIDS displays, and attempts to find all intrusion incidents by correlating different alerts from foreign IP addresses and other traffic information. Specifically, the player should try to label the maximum extent (i.e., all related alerts and pieces of evidence) of all incidents. Players have the opportunity to drill down and explore the dataset within the NIDS to be certain that a threat exists.

2.2 Target Audience

The game, CyFall, is created to study comparisons between network analysts and nonanalyst players.

2.3 Game-Flow Summary

A simple but realistic scenario of observable network traffic evidence that would result from a list of assumed infections from an actual IDS is presented in the game. Observable traffic evidence is essentially a set of “alerts” emitted by the IDS as described above. The possible sequences of using the visual displays are as follows—where A (tabular display), B (parallel coordinate display), and C (node-link display), correspond to the aforementioned visual representations, respectively:

Group 1: A→B→C
Group 2: A→C→B
Group 3: B→A→C
Group 4: B→C→A
Group 5: C→A→B
Group 6: C→B→A

The cyber-security game flow for CyFall is a linear progression through the different visualization displays. See figure 1a for the game overview and figure 1b for the cognitive task process illustration. Six phases make up the entire experience of the game, as described below.

- Phase 1 is the "Introduction Level," where the player(s) is introduced to the purpose of the game and their mission tasks; see figure 2a. The player receives general and relevant information pertaining to the cyber-network data that the visual display it represents. In addition, the way features of the visualization map to represent the network data are described for the player.
- Phase 2 is the "Instruction Level," where the player(s) receives directives and guidance on how to successfully complete their mission tasks. The player(s) also receives a demonstration training that walks them through a practice session of their mission tasks.
- Phase 3 of the "Exercise Level" for Visualization 1 is where the player(s) views the visual representation of cyber-network data and is able to sort, zoom, and take a deeper look into the first visual representation to be presented to them, according to their experimental (group). See figure 2b.
- Phase 4 of the "Exercise Level" for Visualization 2 is where the player(s) views the visual representation of cyber-network data and is able to sort, zoom, and take a deeper look into the second visual representation to be presented to them, according to their experimental (group).
- Phase 5 of the "Exercise Level" for Visualization 3 is where the player views the visual representation of cyber-network data and is able to sort, zoom, and take a deeper look into the third visual representation to be presented to them, according to their experimental (group).
- Phase 6 of the "Epilogue Level" is where the player(s) receives results displaying their overall performance for their mission tasks. It will provide player's accuracy for identifying attacks and intrusion attempts, provide the time elapsed for the completion of each of the three visualization display levels and a combined overall time, and provide their error rate for identifying attacks and intrusion attempts. See figure 2c for an overview of game flow.

Figure 1. (a) Overview of CyFall game flow.

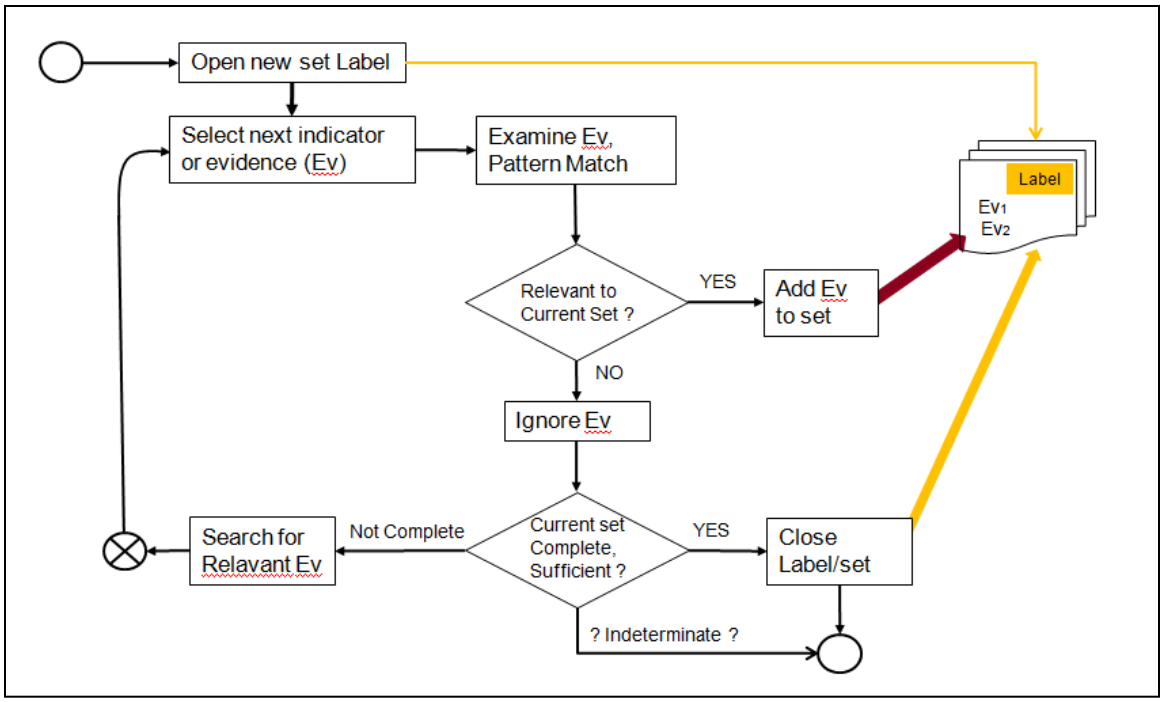

Figure 1. (b) An illustration of the cognitive task analysis process that the player goes through while conducting the assembly of alert-evidence sets.

The three modules: Introduction, Exercise, and Results are coded as an "Introduction Overlay," an "Exercise Overlay," and a "Results Overlay" in the game. The "Introduction Overlay" welcomes the player to the game, provides the player(s) their mission tasks, and gives instructions on how to play the game. The "Exercise Overlay" is where the alert dataset has been adopted into the particular visualizations. The player has the opportunity to explore the display

further to identify threats and intrusion attempts. The “Results Overlay” provides the player(s) with their performance during the game. Figure 2a–c shows screenshots of the mocked-up game flow for CyFall.

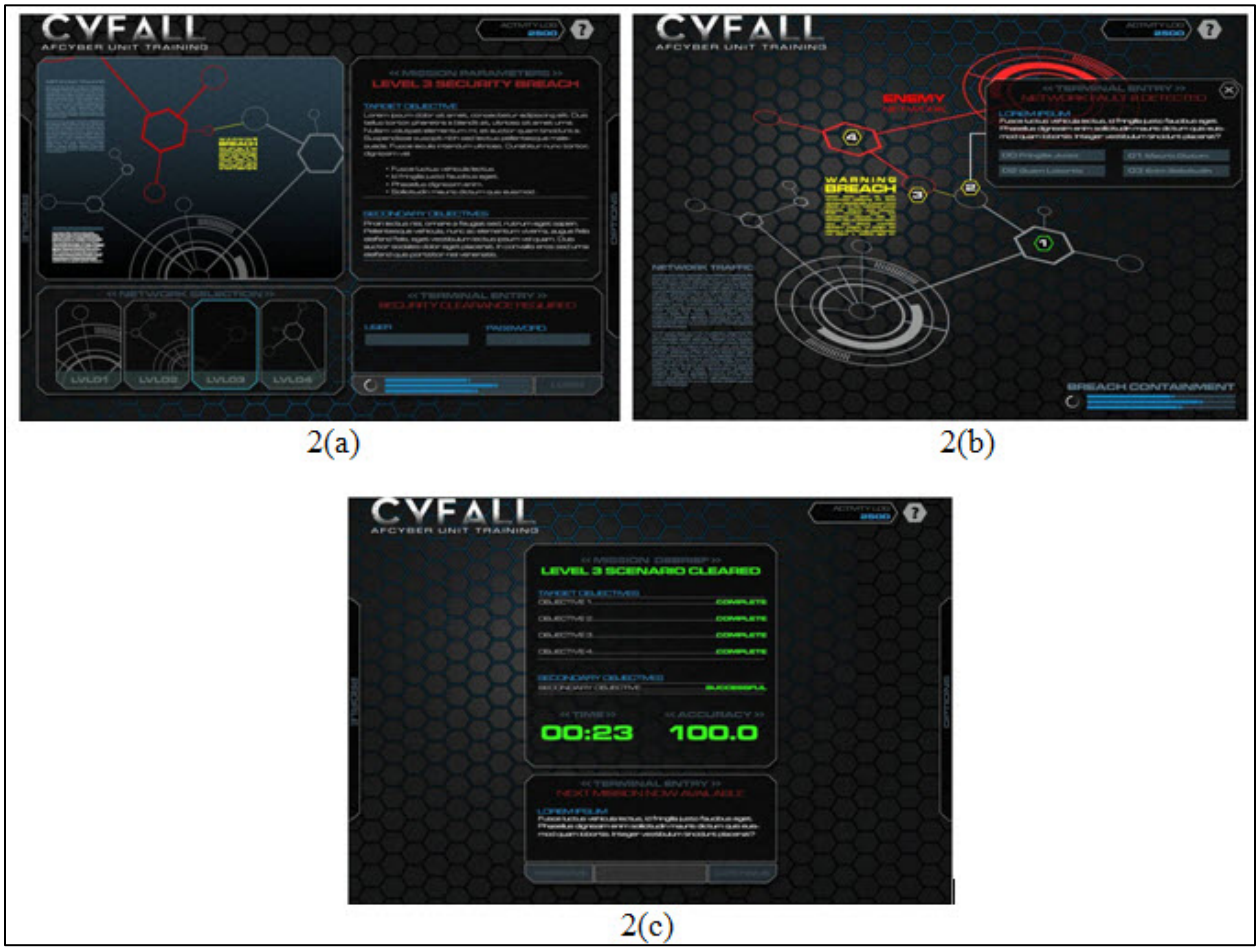

Figure 2. (a) Mocked-up screenshot of the “Introduction Overlay.” (b) Mocked-up screenshot of the “Exercise Overlay.” (c) Mocked-up screenshot of the “Results Overlay.”

3. Game Play and Mechanics

The player’s main actions during the game are as follows for the pattern-matching activity:

Step 1. The player(s) begins the game by logging in with their assigned random identifier identification number.

Step 2. They then enter the “Introduction Overlay” to learn the mission of the game and to complete the training.

Step 3. Next, the player(s) enters into their randomized sequence of the three visual displays,

the “Exercise Overlay.” In the first visual display, they begin to explore the visualization for any evidence or indicators of an attack or intrusion attempt.

Step 4. Player(s) label evidence and indicators when they suspect that an alert is a threat.

Step 5. The player(s) continues the pattern-matching process until they are satisfied that they have identified all possible indicators/evidence that a particular threat exists. This cycle is repeated until all threats and attacks are identified by the player(s).

Step 6. If an alert being observed is not sufficient to the player(s) to support identifying an attack or threat then that evidence/indicator is simply ignored and they move on to the next alert.

Step 7. When “Step 5” is completed to the satisfaction of the player(s), they submit their answers that were labeled and checked off to the game.

Step 8. The player now moves on to the next visual display and begins the process all over again.

Step 9. After completing all visual displays the game will enter into the “Results Overlay,” showing the player(s) their scores, accuracy, time statistics, and overall performance. The game ends.

4. Interface

4.1 Visualization

Visualization refers to a general process whereby a system or tool maps portions of a dataset to graphical symbols and glyphs. The intention is to support the cognitive activities of the player to more easily, or more accurately, form and refine a “mental model” of a situation to optimize situation awareness and decision-making. Traditionally, in cyber security, analysts have used textual displays of features and feature-vectors (records) as the sole vehicle to convey the semantics of the current state. Analysts rely upon each player to interpret this language, attendant semantics, and past (often-substantial) domain experience to create a model of sufficient fidelity to permit decision making. Proponents of visualization assert that by presenting some of the feature dimensions graphically, a more accurate mental model can be obtained, and that it will require less experience to achieve a similar level of decision-making accuracy. From a human computation perspective, the game intends to use the natural parallel processing of the human language understanding and visual perception of subsystems leveraging the human “sense-fusion” capabilities.

A current method used to synthesize a dataset comprising attack alert records, as well as providing some background (normal traffic, or noise) traffic data, is the tabular display. It is a spreadsheet of potential alerts. The benefit of using the tabular display is that all information is easily visible and sortable by the player. Figure 3 shows an example of a tabular display and figure 4 is a proposed tabular display for the game.

	!		Alert ID	Device Type	Duration	Last Change	Device Name	Event Updated	Status
1.			00000YT	Switches and Hubs	0 hr 44 min	29-Jan-2008 14:49:39	fl-69-69-1-2.dhcp.emba...	Utilization	Active
2.			00000YQ	Switches and Hubs	0 hr 44 min	29-Jan-2008 14:49:39	fl-69-69-1-5.dhcp.emba...	Utilization	Active
3.			00000YM	Switches and Hubs	0 hr 44 min	29-Jan-2008 14:49:39	fl-69-69-1-10.dhcp.emb...	Utilization	Active
4.			00000YN	Switches and Hubs	0 hr 44 min	29-Jan-2008 14:49:39	fl-69-69-1-9.dhcp.emba...	Utilization	Active
5.			00000YS	Switches and Hubs	0 hr 44 min	29-Jan-2008 14:49:39	fl-69-69-1-4.dhcp.emba...	Utilization	Active
6.			00000YP	Switches and Hubs	0 hr 44 min	29-Jan-2008 14:49:39	fl-69-69-1-7.dhcp.emba...	Utilization	Active
7.			00000YL	Switches and Hubs	0 hr 44 min	29-Jan-2008 14:49:38	fl-69-69-1-1.dhcp.emba...	Utilization	Active
8.			00000YK	Switches and Hubs	0 hr 44 min	29-Jan-2008 14:49:38	fl-69-69-1-3.dhcp.emba...	Utilization	Active
9.			00000YG	Switches and Hubs	5 days 14 hr	29-Jan-2008 14:49:38	fl-69-69-1-6.dhcp.emba...	Utilization	Active
10.			00000YR	Switches and Hubs	0 hr 44 min	29-Jan-2008 14:49:38	fl-69-69-1-8.dhcp.emba...	Utilization	Active

Figure 3. Example table spreadsheet.

Suspiciou	Date	Time	ToolName	Protocol	SourceEnt	SourceIP	SourcePo	DestEntity	DestIP	DestPort	Country	SrcCC	DstCC	AlertMess	AlertTraff
	10/17/201	15:00:30	Snort	tcp	US1.2	10.234.111	52233	MD0.6	10.66.0.6	52030	MD	US	MD	ET TROJAN	|09 00 00
	10/17/201	15:00:45	Snort	tcp	MD0.6	10.66.0.6	80	US1.2	10.234.111	52200	MD	MD	US	ET TROJAN	PONG |3a|
☑	10/17/201	15:00:50	Snort	tcp	US1.2	10.234.111	81	MD0.6	10.66.0.6	52330	MD	US	MD	ET TROJAN	|09 00 00
	10/17/201	15:01:10	Snort	tcp	US1.2	10.234.111	52001	MD0.6	10.66.0.6	80	MD	US	MD	ET TROJAN	GET|20|/|
	10/17/201	15:01:35	Snort	tcp	US1.2	10.234.111	52001	MD0.6	10.66.0.6	80	MD	US	MD	ET TROJAN	
☑	10/17/201	15:59:05	Aggregate	tcp	US1.3	10.234.111	*	US.10	10.250.10.	80	US	US	US	High traffic	*
	10/17/201	15:20:12	Snort	tcp	US1.5	10.234.111	51119	US.10	10.250.10.	80	US	US	US	ET TROJAN	GET /instal
	10/17/201	15:59:00	Aggregate	tcp	US1.5	10.234.111	*	US.10	10.250.10.	80	US	US	US	High traffic	*
	10/17/201	15:32:15	Snort	tcp	US.55	10.234.111	80	US.10	10.250.10.	80	US	US	US	ET TROJAN	GET /instal
☑	10/17/201	15:45:00	Aggregate	tcp	US.55	10.234.111	*	US.10	10.250.10.	80	US	US	US	High traffic	*
	10/17/201	15:45:50	Snort	tcp	US0.2	10.123.100	55500	US.10	10.250.10.	80	US	US	US	ET TROJAN	GET /instal
	10/17/201	15:59:00	Aggregate	tcp	US0.2	10.123.100	*	US.10	10.250.10.	80	US	US	US	High traffic	*

Figure 4. The suggested tabular visual display created using Java* script and implemented in an online open-source survey program called LimeSurvey.

It is customary for a graphical visualization to provide some background and normal traffic often as a separate layer, because it is often desirable to compare the set of alerts to the much more voluminous normal traffic. Normal traffic must be consistent with the scenario while being truly representative of normal benign traffic in all other aspects. We chose a suitable set of flow-file data and remapped all IP addresses from the capture ranges to the scenario. This resulted in an auxiliary subgraph that can be used to start a scenario graph; the nodes and edges are thus, added

*Java is a trademark of Oracle.

to the graph during parsing of the input-alert file and then the entire composite graph is emitted for the particular display.

One of the graphical representations suggested for the game is a parallel coordinate display. A parallel coordinate graphical representation plots the relationship of mapped rows in a data table as a line. An attribute of a row is represented by a single point on the line (*21*). Box plots are graphical representations of statistical measures—such as, median and lower quartiles, minimum and maximum data values (*22*). We superimpose the box plots onto the parallel coordinate graphical representation and together make a visualization display. For the game, we recommend placing the friendly axis and communications port on the left of the display and placing the hostile, or unfamiliar axes, on the right side of the display. For the example, we grouped the hostile axis such that nodes are shown in the nonoverlapping country range-boxes (superimposed box plots) in figure 5.

Our preferred graphical rendering and analysis environment is GUESS (*23*)—an exploratory data analysis and visualization tool for graphs and networks—that requires an input file format comprised of sections, first defining the vertices (nodes) and then the edges (links). A Perl 5 (*24*) script was used to parse the scenario file of alerts in CSV file format to construct an internal representation of the graph and required attributes. The final procedure in the script then retrieved the graph—nodes first, then edges—and at the same time, performed spatial location and colorization mapping to create a graph that incorporates the desired display metaphor (parallel coordinate, aggregate/situational awareness). We used:

- 5parallelCoords.pl to generate a parallel coordinates layout with five vertical axes.
- Vis3.pl to generate the aggregated layout for use as the visual representation for the game.

The illustrations for figures 5–7 were created using GUESS (*23*). The line weights and colors are random and are limited by what can be generated with GUESS. An improvement with a range of colors and weight symbols is strongly encouraged. These nodes represent internal servers used and owned by the friendly nodes; however, some are attacked as well, but shown in a separate area as typical enterprise servers. We recommend additional improvements that add node sizes, shapes, and colors for easier selection in the viewing environment. The country polygons are to be color coded and labeled with the two-character country codes provided in the dataset. See figure 6a and 6b for the initial parallel coordinate graphical representation generated by GUESS. The corrected generated graphs in figure 7a and 7b remove the out of place two blue vertical lines that are supposed to be horizontal.

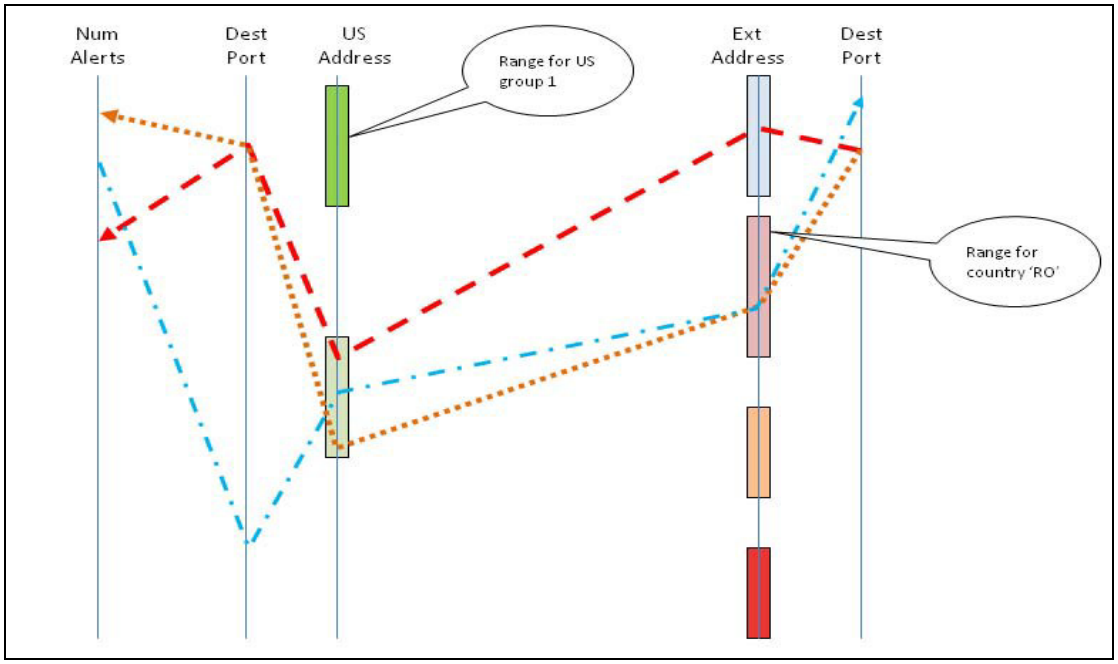

Figure 5. Parallel coordinate metaphor with port and number of alerts attribute axes.

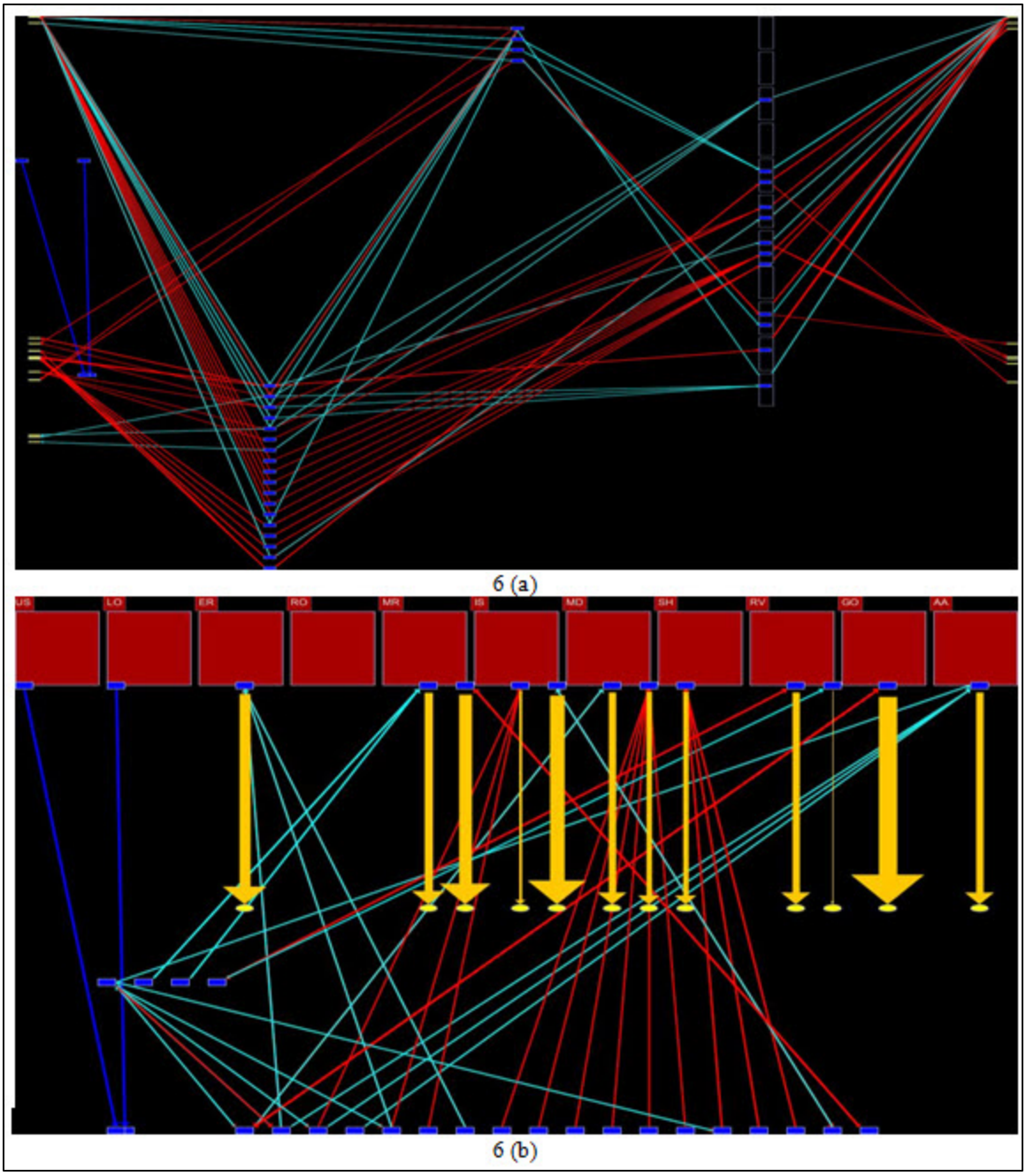

Figure 6. (a) The original generated graph from the GUESS visualization tool. An error exists in the data that produced two blue out of place vertical lines that are supposed to be horizontal. This is fixed in figure 7a. (b) The original generated graph from the GUESS visualization tool. The background color was too dark and needs to be lightened. Also, the country codes are hard to see on top of the red squares. Figure 7b is the updated selection of the parallel coordinate display.

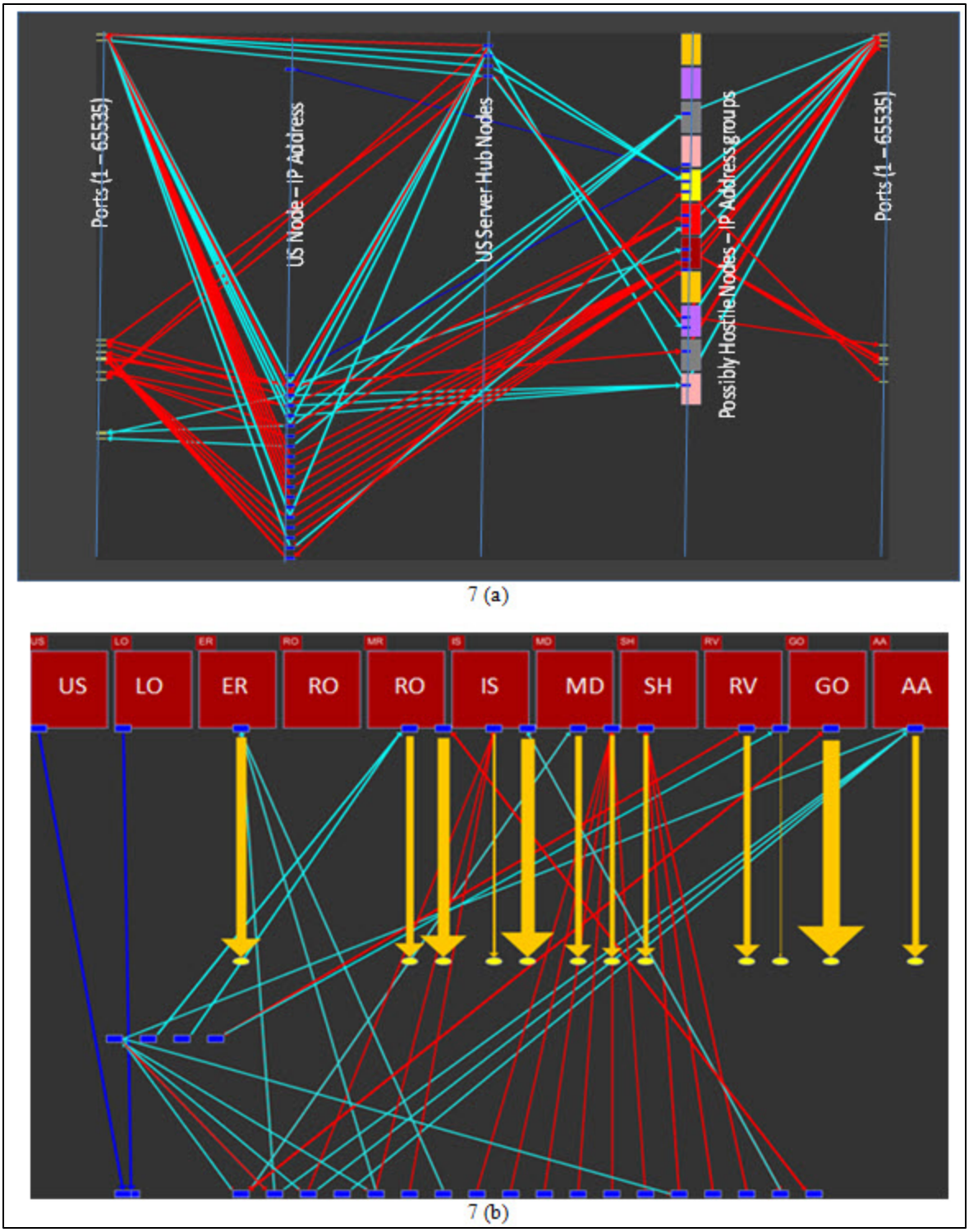

Figure 7. (a) Corrected parallel coordinate display generated by GUESS for the game. (b) The updated selection of the parallel coordinate display generated by GUESS for the game with a lighter background color and larger country-code names.

The second graphical representation suggested for the game is node-link display. A node is a vertex or point within a graph (*25*). It helps to illustrate relationships with other objects when connected by links. Links are edges or arcs that connect nodes together (*26*). In figure 8a, we show a possible implementation of the aggregated visualization. Here glyph areas represent the countries. The individual alerts from each unfriendly node have been aggregated into one directed arrow with its width proportional to the number of alerts it represents. In use, initially, only the aggregated arrows would be shown, and allow the hostile node to be selected to generate a detail popup for incident labeling. It is possible that the detailed arrows (alerts) could be selected for display as an additional layer. See figure 8a and 8b for variations of the aggregated node-link display in grayscale and in color.

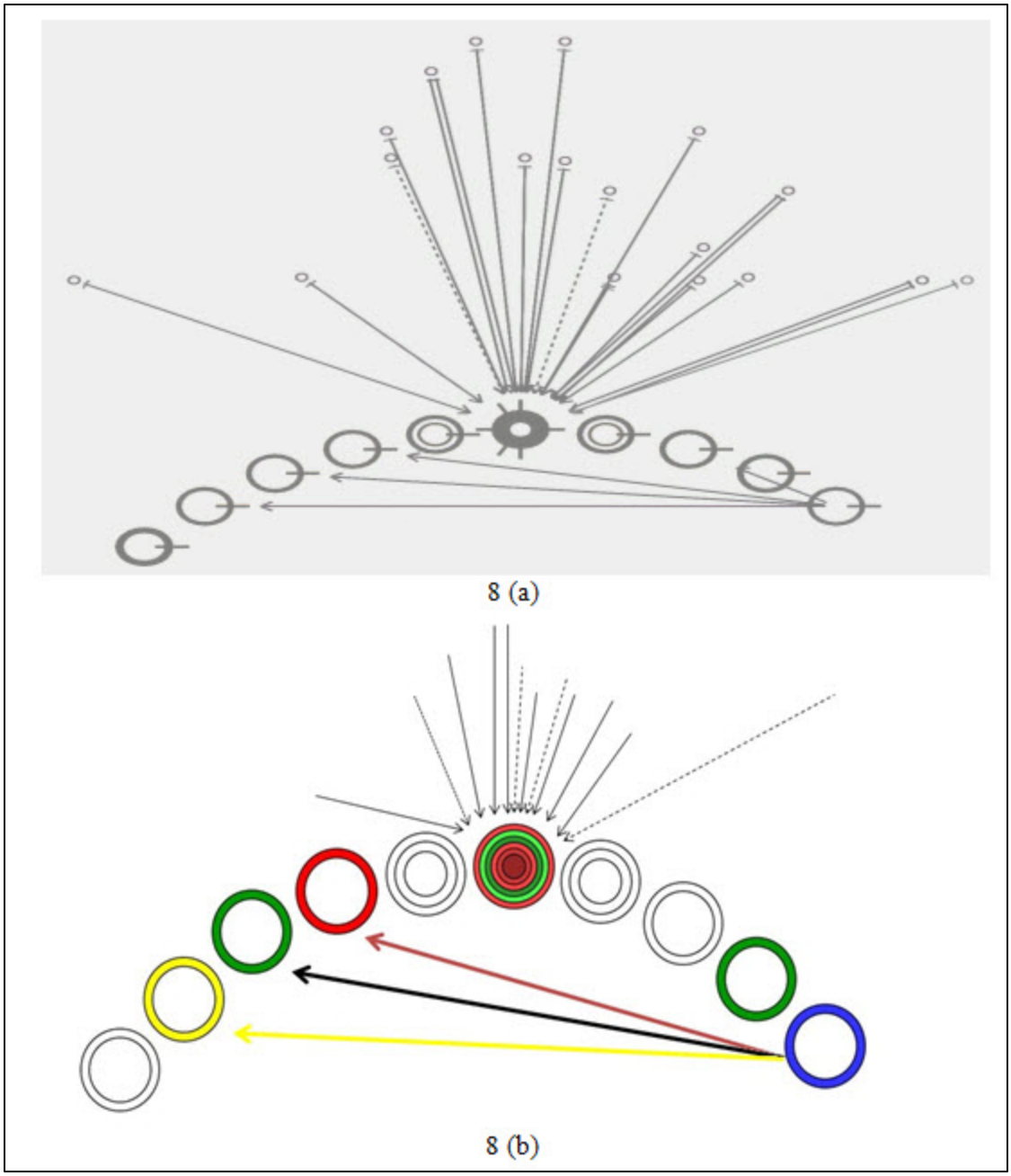

Figure 8. (a) Original node-link visual representation and idea for the game. (b) Node-link graphical representation with suggested color-coding scheme to represent the dataset.
Note: See figure 9 for further details about the color scheme as well as additional symbols that can be used to visualize the data. See figure 10 for additional potential link representation.

4.2 Visualization Attributes

We have suggested node attribute variations that may be used to represent various entities within the cyber-security network paradigm. For example, the node symbols are used in the game to represent workstations, servers, and printers that might exist in the United States as well as in other countries.

See figure 9 below for the nodal representations used in the game—particularly for the aggregated visual representation. Similarly, the fabricated attributes suggested for the link representations for the aggregated display are presented in figure 10.

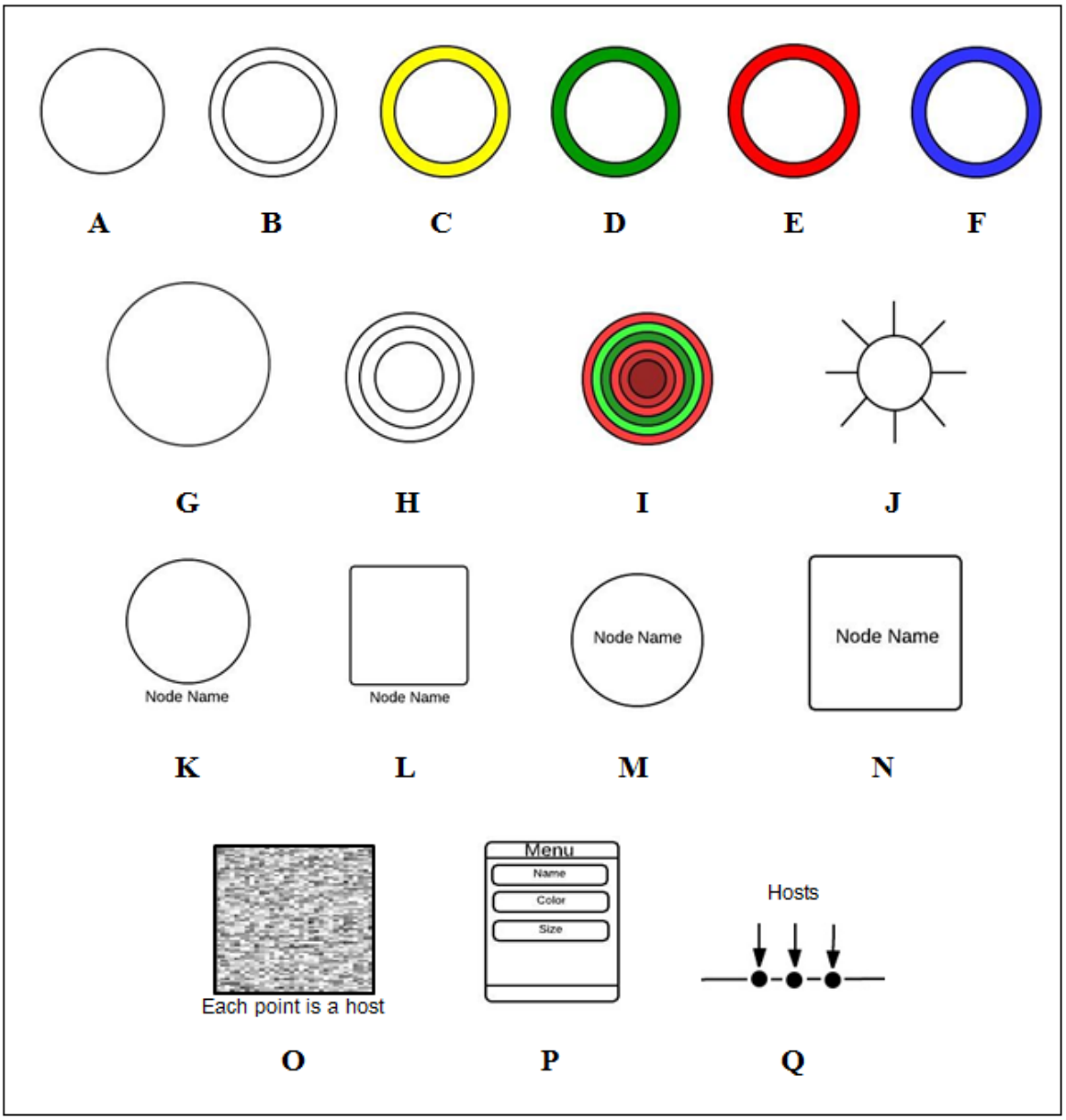

Figure 9. Fabricated node (graph vertex) representations for CyFall's aggregated display.

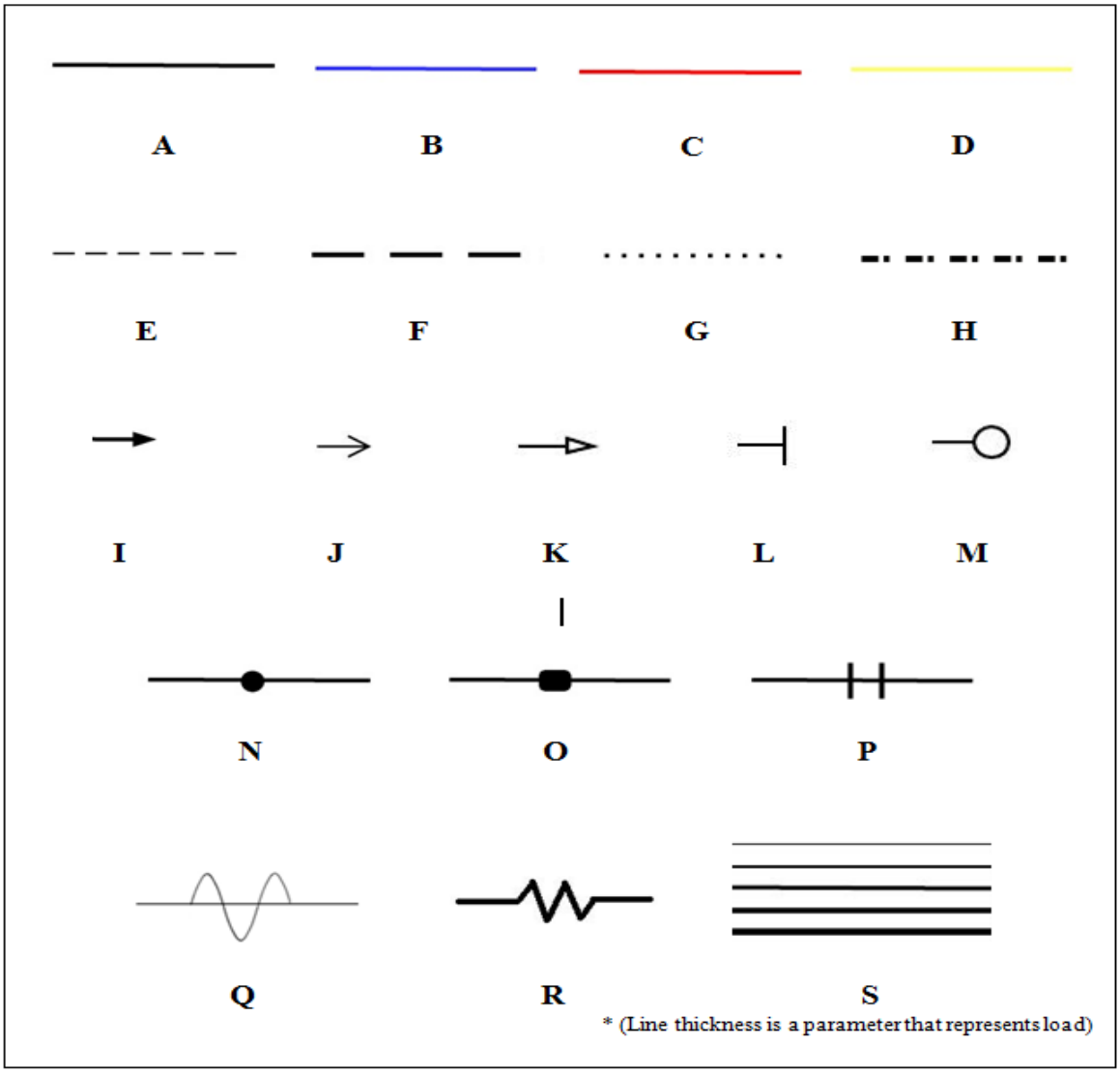

Figure 10. Fabricated link (graph edge) representations for CyFall.

4.3 Player Interaction with Tabular Display

We recommend using Zoho (*27*), a Web-based online office suite of applications used for productivity and collaboration. Zoho allows spreadsheets to be uploaded or created and provides a scrolling presentation with column sort functions. An interface mechanism to allow checkbox invoked labeling with set number increment and decrement is suggested to complement Zoho's capabilities. The other two visual representations in addition to the tabular representation would be displayed in the recommended randomized sequence. Figures 11, 12, and 13 highlight some of the features when using Zoho.

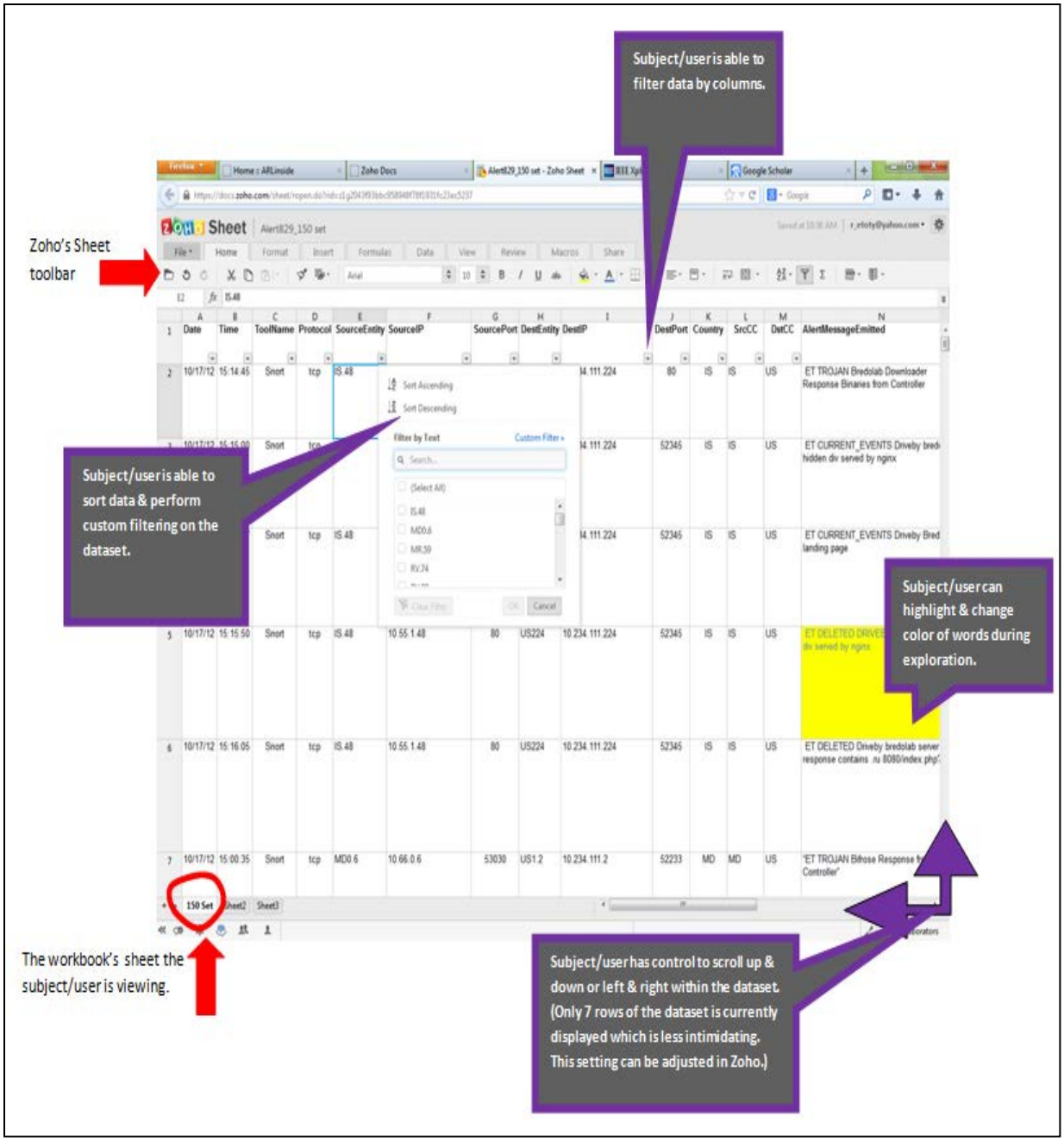

Figure 11. Screenshot of Zoho's interactive table.

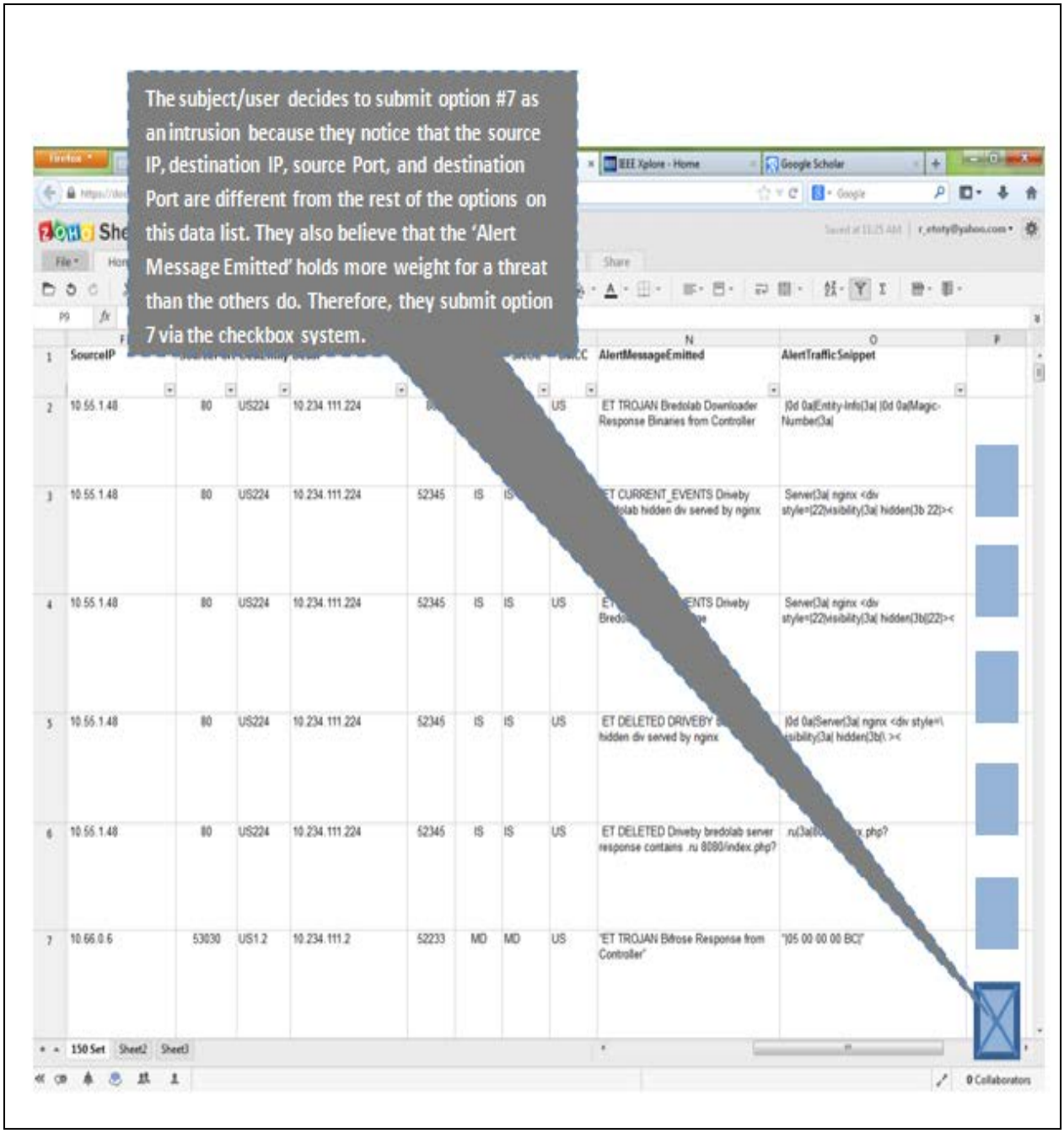

Figure 12. Illustration of checkbox idea.

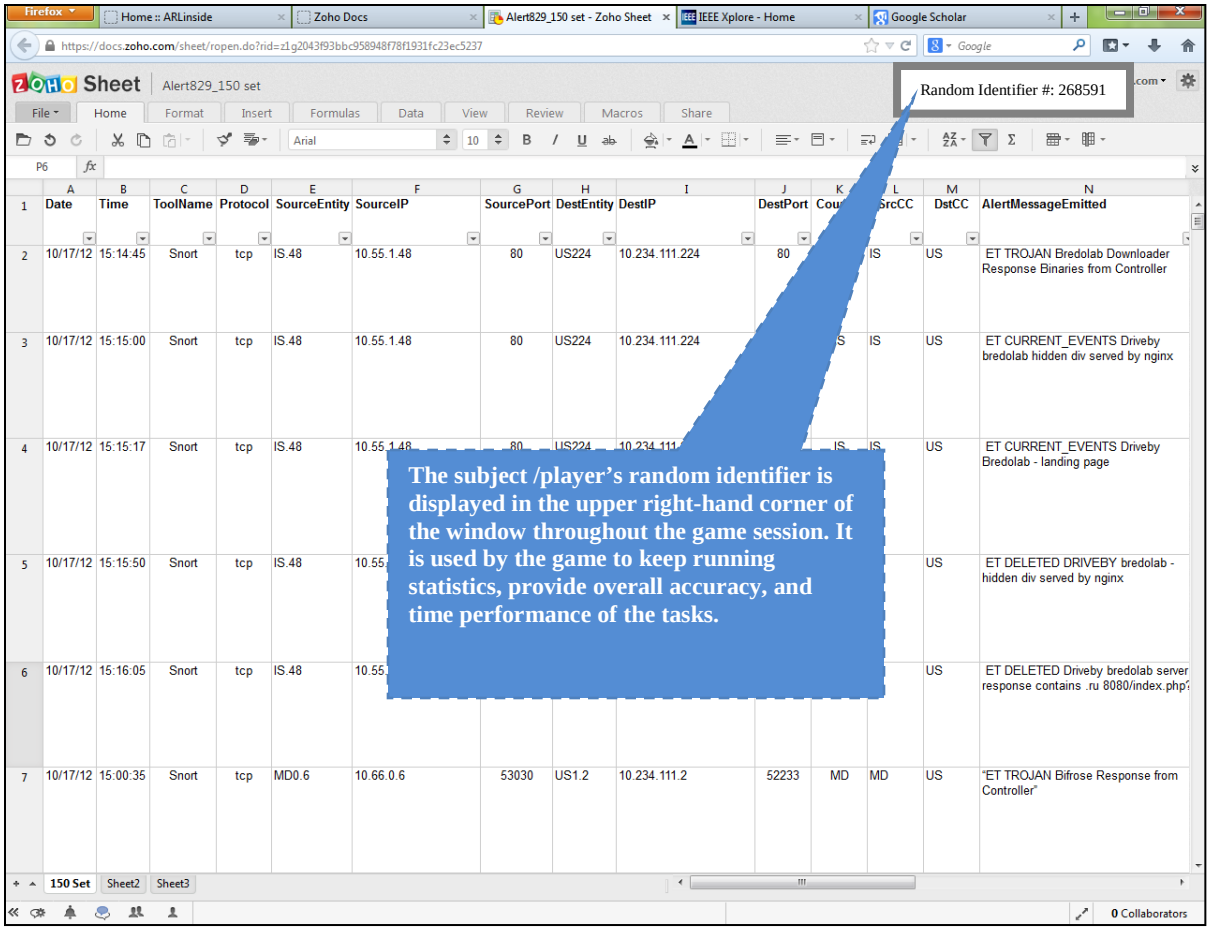

Figure 13. Illustration of the player's random identifier usage idea.

4.4 Player Interaction with Tabular Display

The parallel coordinate and node-link displays are graphical representations of computer network alerts. CyFall allows the player(s) the opportunity to click on the graph's attributes for further exploration. Hence, the player(s) is able to probe into the display to obtain additional information to help them identify what appears to be an attack or an intrusion. This method is implemented with reverse and hot-spot lookup tables. Regions of the visualization determined by the player to be significant threats—specifically, implying attacks (True Positive [TP]) and intrusion attempts (False Positive [FP])—are logged into the game by clicking near a particular graph feature.

4.5 Hardware Requirements

The target hardware for the CyFall game is to be displayed on a wall using a projector machine. The player(s) will be confined in a scanner machine (i.e., an EEG or fMRI machine) where their methods of interaction with the game will be via mouse or keyboard. A small window on the scanner machine allows the player to view the game on a screen located outside of the machine. It is on this screen where the game is being projected from the projection machine. No Internet connection is required to play the game.

5. Story, Setting, and Character

5.1 Background and Storyline

It was called “The Shift.” In 2016, a massive discharge of energy in the upper atmosphere of earth tore a hole in the fabric of space-time. Without warning, a large swath of the contiguous United States of America was “shifted” from what is now called the “Prime” dimension to the “Arda” dimension. Cities, people, towns, farms, animals … all were moved to a completely new world on a completely new continent, “Middle Earth.” Chaos ensued, as the placements of objects from one dimension were indiscriminately determined for deposition within Middle Earth. This came during a difficult time when those who were identified as “good” within Arda had recently defeated a great “evil,” Sauron, the master of Mordor.

Over the course of a decade, the chaos subsided and the inhabitants from Prime eventually established order within the land their country had settled on. This period became known as the “Great Edification.” Humans from Prime began to organize and interact with the indigenous human populations and soon discovered that they were not the only species to inhabit this land. Along with humans, Middle Earth was also inhabited by elves, dwarves, orcs, goblins, ents, and halflings—along with numerous other unidentified species; each bearing allegiance to their own various nations.

A year has passed since “Operation Screaming Nazgul,” a nonphysical military attack on the U.S. through “Middlenet” itself. As a result, a communications infrastructure was built to facilitate information sharing, an initiative between the several nation states. The player is a cyber analyst for “Shinning Glamdring.” As an analyst, they are responsible for utilizing various intrusion detection tools in order to stop or hamper future cyber attacks against the U.S. In this specific scenario, they will be analyzing the alerts from various intrusion sensors utilized throughout the U.S. network. Their job will be to correlate various alerts and determine when a legitimate threat is imminent. Like most alert tools, not every detection mechanism is foolproof. The player is presented alert information in three different formats:

- Tabular display,
- Parallel coordinates superimposed with box-plots display,
- Nodal-link display.

Using these views, interpret the information presented and identify potential evidence that an attack is, or has occurred, while dismissing any identified false positives.

Performing this duty is imperative. Our country must be protected from the widespread chaos after “Operation Screaming Nazgul.” Furthermore, it is believed Sauron may have survived his

last battle, and now his power and influence are growing again. It becomes imperative that the “Prime Project” be completed in order to secure possible reinforcements from Prime if Sauron reawakens to his former power. Hence, the countries (nation states) that communicate friendly or unfriendly with the United States are:

- Shire
- Rivendell
- Lothlorian
- Eriador
- Gondor
- Rohan
- Mordor
- Moria
- Isengard

We used country codes to represent a country’s source IP address, destination IP address, and destination port. The parameters were used to fabricate the graphical representations within the visual displays.

6. References

1. Rouse, M. SearchSecurity. Last Updated: October 2006. http://searchsecurity.techtarget.com/definition/hacker (accessed on 09/25/2013).
2. Rouse, M. SearchSecurity. Last Updated: June 2007. http://searchsecurity.techtarget.com/definition/cracker (accessed on 09/25/2013).
3. Rouse, M. SearchSecurity. Last Updated: October 2008. http://searchsecurity.techtarget.com/definition/HIDS-NIDS (accessed on 09/25/2013).
4. Wikipedia. Last Updated: May 6, 2014. http://en.wikipedia.org/wiki/Firewall_%28computing%29 (accessed on 09/25/2013).
5. Gil, P. About.com. http://netforbeginners.about.com/od/a/g/antivirus.htm (accessed on 09/25/2013).
6. Rouse, M. SearchSecurity. Last Updated: October 2006. http://searchsecurity.techtarget.com/definition/spyware (accessed on 09/25/2013).
7. SANS. http://www.sans.org/security-resources/idfaq/what_is_hips.php (accessed on 09/25/2013).
8. Rights, Retains Full. SANS Institute InfoSec Reading Room (2002).
9. Google. http://www.garlic.com/~lynn/secure.htm (accessed on 09/24/2013).
10. Gerth, J. *Proceedings of the Seventh International Symposium on Visualization for Cyber Security, ACM*, New York, NY, USA, 2010.
11. 98–17, Dept. of Computer Eng. Chalmers Univ. of Tech, SE-412 96 Goteborg, Sweden, December 1998. URL: http://www.ce.chalmers.se/staff/sax.
12. Monchi, O.; Petrides, M.; Petre, V.; Worsley, K. J.; Dagher, A. Distinct Neural Pathways Activated During Four Stages of the Wisconsin Card Sorting Task Using Event-Related fMRI, *NeuroImage*, *13*(6), 448, 2001.
13. Ellis, G.; Dix, A. An Explorative Analysis of User Evaluation Studies in Information Visualization. In *Proceedings of the 2006 AVI Workshop on Beyond Time and Errors: Novel Evaluation Methods for Information Visualization (BELIV '06)*, ACM, New York, NY, USA, 1–7, 2006.
14. Erbacher, R. F.; Frincke, D. A.; Moody, S. J.; Fink, G. A Multi-Phase Network Situational Awareness Cognitive Task Analysis. *Information Visualization Journal*, (pp. 204–219), 2010.

15. Hiraishi, H.; Mizoguchi, F. Design of a Visual Browser for Network Intrusion Detection. In *Enabling Technologies: Infrastructure for Collaborative Enterprises, 2001. WET ICE 2001. Proceedings. Tenth IEEE International Workshops* (pp. 132–137) IEEE, 2001.

16. Becker, R. A.; Eick, S. G.; Wilks, A. R. Visualizing Network Data. *Visualization and Computer Graphics, IEEE Transactions* 1995, *1* (1), 16–28.

17. Abdullah, K.; Lee, C.; Conti, G.; Copeland, J. A. Visualizing Network Data for Intrusion Detection. In *Information Assurance Workshop, 2005. IAW'05. Proceedings from the Sixth Annual IEEE SMC* (pp. 100–108). IEEE, June 2005.

18. Xin, J.; Dickerson, J. E.; Dickerson, J. A. Fuzzy Feature Extraction and Visualization for Intrusion Detection. In *Fuzzy Systems, 2003. FUZZ'03. The 12th IEEE International Conference* (Vol. 2, pp. 1249–1254). IEEE, May 2003.

19. Emerging Threats. (n. d.) Emerging-All Rules. Retrieved from Emerging Threats: http://rules.emergingthreats.net/open/snort-2.9.0/emerging-all.rules (accessed 09/23/2013).

20. Schmitz, C. Limesurvey – The Open Source Survey Application. Hamburg [cited 31.03. 2009] Available from: http://www. Limesurvey.org. External link (2009).

21. What is a Parallel Coordinate Plot? http://stn.spotfire.com/spotfire_client_help/para/para_what_is_a_parallel_coordinate_plot.htm (accessed on 09/23/2013).

22. Easycalculation. http://easycalculation.com/maths-dictionary/boxplot.html (accessed on 09/23/2013).

23. Adar, E. GUESS: A Language and Interface for Graph Exploration. *CHI 2006*, Montreal, Canada, April 22–27, 2006, pp. 791–800.

24. Jon Allen (JJ). perldoc.perl.org Perl Programming Documentation. http://perldoc.perl.org/perl.html (accessed on 10/11/2013).

25. Rouse, M. SearchNetworking. Last Updated: August 2006. http://searchnetworking.techtarget.com/definition/node (accessed on 09/23/2013).

26. Caldwell, C. Graph Theory Glossary. Copyright 1995. http://www.utm.edu/departments/math/graph/glossary.html (accessed on 09/23/2013).

27. Zoho. Copyright 2013. http://www.zoho.com/ (accessed on 09/09/2013).

28. D'Amico, A.; Whitley, K. The Real Work of Computer Network Defense Analysts. *VizSEC 2007: Proceedings of the Workshop on Visualization for Computer Security*, Berlin; Heidelberg: Springer-Verlag, pp. 19–37, 2008.

INTENTIONALLY LEFT BLANK.

List of Symbols, Abbreviations, and Acronyms

ARL	U.S. Army Research Laboratory
CPU	central processing unit
CSV	comma-separated values
DDoS	distributed denial-of-service
FMRI	functional magnetic resonance imaging
FP	False Positive
HIDS	host-based intrusion detection systems
ID	intrusion detection
IDS	intrusion detection system
IP	Internet Protocol
NIDS	network intrusion detection system
TP	True Positive
WCST	Wisconsin Card Sorting Task

1 (PDF) DEFENSE TECHNICAL INFORMATION CTR DTIC OCA

2 (PDF) DIRECTOR US ARMY RSRCH LAB RDRL CIO LL IMAL HRA MAIL & RECORDS MGMT

1 (PDF) GOVT PRINTG OFC A MALHOTRA

1 (PDF) US ARMY RSRCH LAB ATTN RDRL CIN D R ETOTY